CS5691 : Pattern Recognition and Machine Learning

Assignment 2 Report

Group No. 23

Group Members :
Aakriti Budhraja (CS18S009)
Madhura Pande (CS17S031)
Sadbhavana Babar (CS18S029)

Course Instructor:
Prof. C. Chandra Sekhar
Professor, Dept. of CSE,
Indian Institute of Technology, Madras

October 31, 2018

Abstract

The goal in classification is to take an input vector x and to assign it to one of the K discrete classes C_k where k = 1,2,....K. The classes are taken to be disjoint, so that each input is assigned to one and only one class. This report talks about various ways of Pattern classification that we have performed on the following datasets:

- Dataset 1 - Two-dimensional artificial data :
  - Linearly separable dataset for static pattern classification
  - Non-linearly separable dataset for static pattern classification
  - Overlapping dataset for static pattern classification
- Dataset 2 - Real world data :
  - Dataset for static pattern classification
  - Image dataset for varying length pattern classification

Classifiers used are as follows :

- K-Nearest Neighbour classifier
- Bayes and Naive Bayes classifier with a Gaussian Distribution for each class
- Bayes, Naive Bayes classifier with a Gaussian Mixture Model for each class
- Bayes Classifier with K-Nearest Neighbours method for estimation of class-conditional probability density function

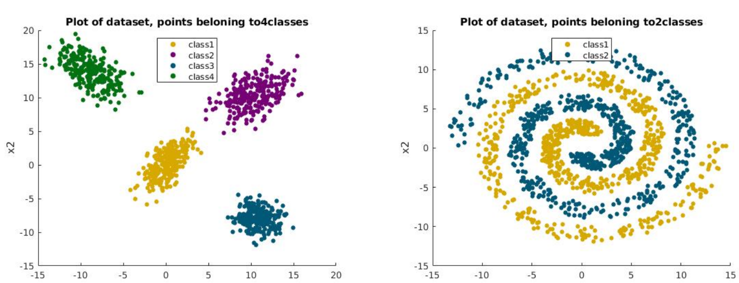

Figure 1: Linearly separable and Non-Linearly seperable data

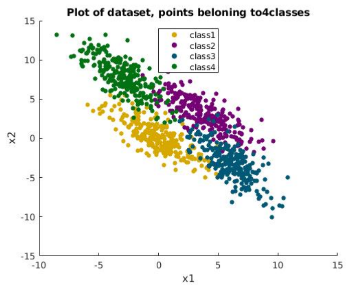

Figure 2: Overlapping data

1 K Nearest Neighbours

$$K = \sum_{i=1}^{M} Ki$$

$$i^* = argmax_i(K_i)$$

$K_i =$ Number of training samples of class i among the K Nearest Neighbours.

Approach: Given any new sample x, we find its K nearest neighbours. Then, we see which class has the maximum number of such neighbours. That class gets assigned to x.

1.1 Linearly Separable Data

Figure 3: Best model : 1-NN

Observations:

- For k = 1, accuracy of 100% is obtained on train, test and validation datasets.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

1.2 Non-linearly Separable Data

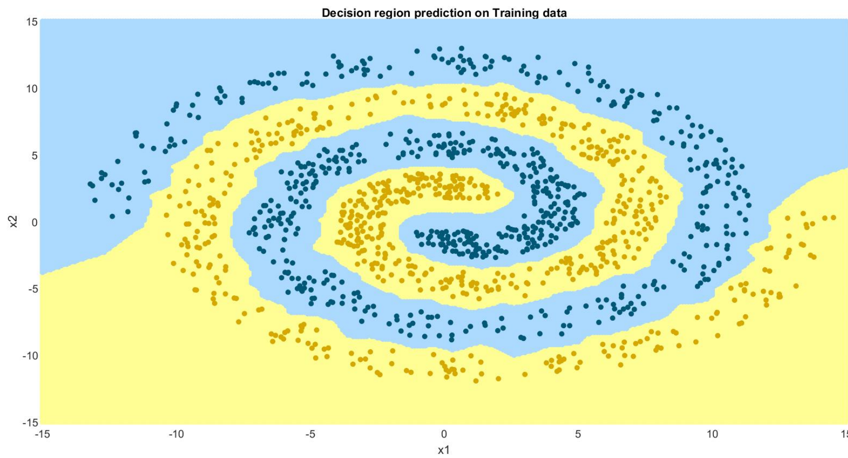

Figure 4: Best model : 1-NN

Observations:

- For k = 1, accuracy of 100% is obtained on train, test and validation datasets.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

1.3 Overlapping Data

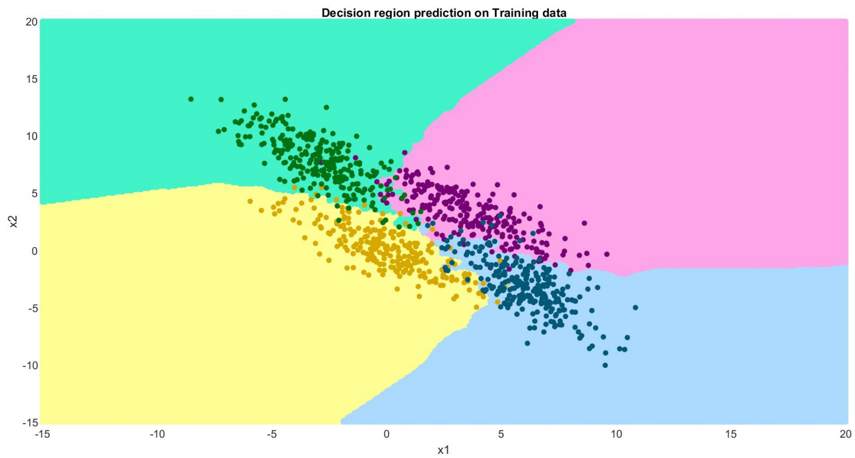

Figure 5: Best model : 5-NN

Table 1: Classification accuracies of model

k	**1**	**2**	**3**	**4**	**5**	**6**
$TrainData$	100	95	95.20	94	93.90	93.70
$ValidationData$	90.67	89.83	92.16	91.83	93	92.67

Observations:

- The best model is 5-NN model with an accuracy of 93% on Validation Dataset.
- Accuracy of best model on Test Data is 90.5%.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 93.6000 & 0.4000 & 3.2000 & 2.8000 \\ 0 & 92.0000 & 4.4000 & 3.6000 \\ 1.2000 & 4.0000 & 94.8000 & 0 \\ 2.4000 & 2.4000 & 0 & 95.2000 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 91 & 0 & 5 & 4 \\ 0 & 91 & 4 & 5 \\ 4 & 10 & 86 & 0 \\ 3 & 3 & 0 & 94 \end{bmatrix}$$

2 Designing Naive-Bayes and Bayes classifiers

2.1 Naive-Bayes classifier with a Gaussian distribution for each class

One of the most useful ways to represent pattern classifiers is in terms of a set of discriminant functions given by

$$g_i(\mathbf{x}) \tag{1}$$

for i = 1,2, ...C where C is the number of classes. The classifier is said to assign a feature vector x to class $\mathrm{y_i}$ if

$$g_i(\mathbf{x}) > g_j(\mathbf{x}), \forall\, i \neq j \tag{2}$$

Thus, the classifier is viewed as a network or machine that computes C discriminant functions and selects the category corresponding to the largest discriminant. The class label i^* is assigned to a test example such that

$$i^* = argmax_i \; g_i(\mathbf{x}) \tag{3}$$

Here we are trying to maximize the likelihood where $\mathrm{g}_i(\mathbf{x})$ can take any of the following forms :

$$P(y_i/x) \quad = p(\mathbf{x}/y_i)P(y_i) = ln(p(\mathbf{x}/y_i)P(y_i)) \tag{4}$$

Here we have a Gaussian distribution for each class. Analysis is done on dataset 1 for different types of covariance matrices and we predicted the decision regions for a Naive-Bayes classifier. Following are the cases considered :

- Case I : Covariance matrix for all classes is same and is σ^2I
- Case II: Covariance matrix for all classes is same and is C
- Case III: Covariance matrix for each class is different

2.1.1 Linearly separable data

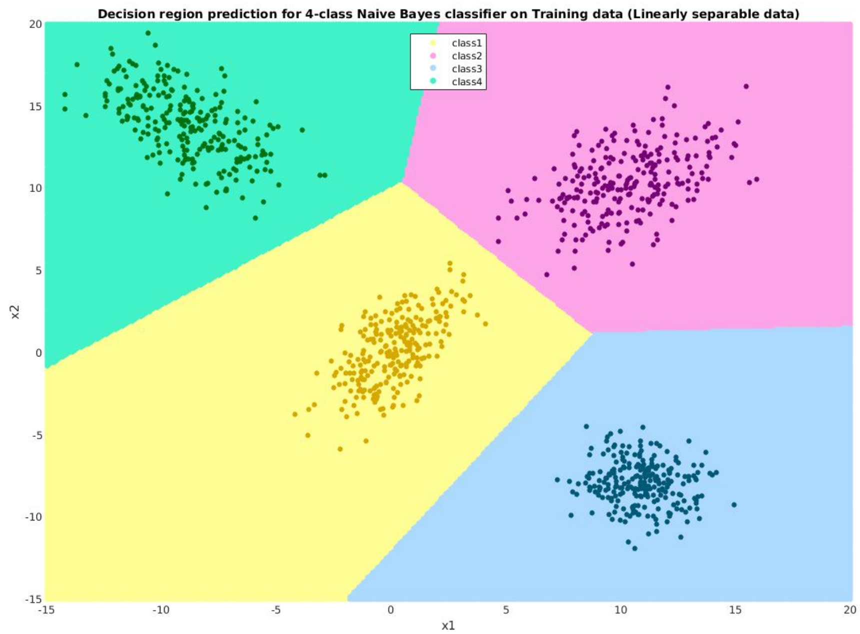

Figure 6: Best Model obtained when the covariance matrix is same

Table 2: Classification accuracies of the model for different values of Covariance matrix

Covariance matrix	$\mathbf{C} = \sigma^2\mathbf{I}$	**same C**	**different C**
$TrainData$	100	100	100
$ValidationData$	100	100	100

Observation:

- Classification accuracy on test data for all the three cases is same which is 100%

- The decision regions for cases I and II are both separated by linear decision boundaries, whereas the decision boundary for case III is quadratic.
- So to eliminate the model complexity, we considered linear decision boundary as a choice for the best model where the covariance matrix is same and is C.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

2.1.2 Non-linearly separable data

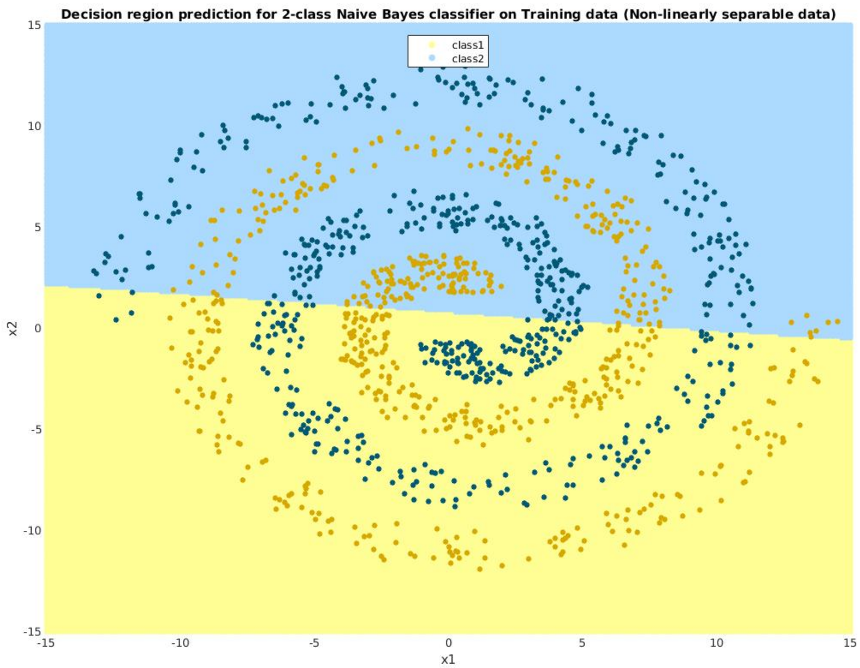

Figure 7: Best model is obtained when the covariance matrix is same

Table 3: Classification accuracies of the model for different values of Covariance matrix

Covariance matrix	$\mathbf{C} = \sigma^2\mathbf{I}$	**same C**	**different C**
$TrainData$	35.58	35.43	55.21
$ValidationData$	57.16	57.29	56.91

$Observation$:

- Classification accuracy on test data for the best model is 53.65%

- The decision region boundaries for cases I, II and III all are linear as the original covariance matrices for both the classes turned out to be almost equal.
- We chose the best model for which the classification accuracy on validation data is maximum which is 57.29% when the covariance matrix is same and is C.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 49.8466 & 50.1534 \\ 78.9877 & 21.0123 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 58.0769 & 41.9231 \\ 50.7692 & 49.2308 \end{bmatrix}$$

2.1.3 Overlapping data

Figure 8: Best model obtained when the covariance matrix is different

Table 4: Classification accuracies of the model for different values of Covariance matrix

Covariance matrix	$\mathbf{C} = \sigma^2\mathbf{I}$	**same C**	**different C**
$TrainData$	86.10	86.10	86.40
$ValidationData$	89.25	89.33	89.67

Observation:

- Classification accuracy on test data for the best model is 89%
- The decision region boundary for cases I and II is linear whereas for case III, the decision region boundary is quadratic.
- We chose the best model for which the classification accuracy on validation data is maximum which is 89.67% when covariance matrix is different.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 83.60 & 0.80 & 8.40 & 7.20 \\ 0 & 81.60 & 13.60 & 4.80 \\ 3.20 & 9.60 & 87.20 & 0 \\ 3.60 & 3.20 & 0 & 93.20 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 86 & 0 & 10 & 4 \\ 0 & 88 & 6 & 6 \\ 4 & 7 & 89 & 0 \\ 6 & 1 & 0 & 93 \end{bmatrix}$$

2.2 Bayes classifier with a Gaussian distribution for each class

Approach:
Analysis is done on dataset 1 for different types of covariance matrices and we predicted the decision region boundary for a Bayes classifier. Following are the cases considered :

- Case I: Covariance matrix for all classes is same and is C
- Case II: Covariance matrix for each class is different

2.2.1 Linearly separable data

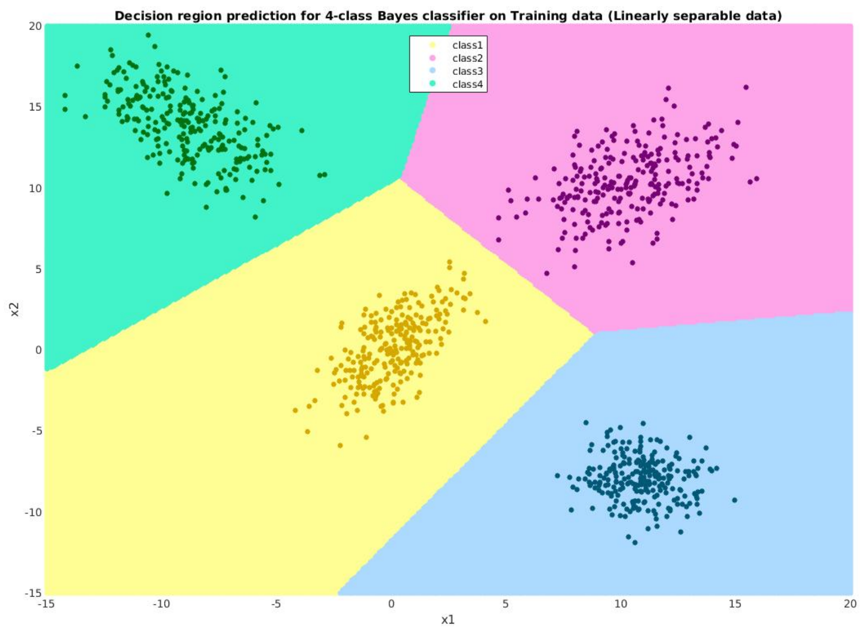

Figure 9: Best model obtained when the covariance matrix is same

Table 5: Classification accuracies of the model for different values of covariance matrix

Covariance matrix	**same C**	**different C**
$TrainData$	100	100
$ValidationData$	100	100

Observation:

- Classification accuracy on test data for both the cases is same which is 100%
- The decision region boundary for case I is linear whereas the decision region boundary for case 2 is quadratic.
- So to eliminate the model complexity, we considered linear decision boundary as a choice for the best model where the covariance matrix is same and is C.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

2.2.2 Non-linearly seaprable data

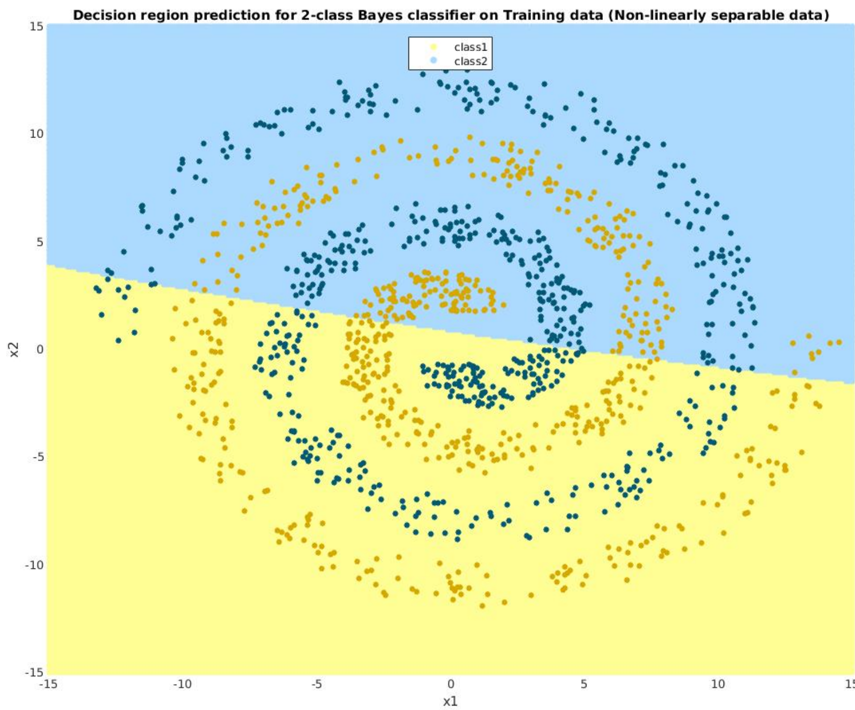

Figure 10: Best model obtained when the covariance matrix is different

Table 6: Classification accuracies of the model for different values of covariance matrix

Covariance matrix	**same C**	**different C**
$TrainData$	35.42	54.98
$ValidationData$	57.16	57.54

$Observation$:

- Classification accuracy on test data for the best model is 53.46%

- The decision region boundaries for cases I, II and III all are linear as the original covariance matrices for both the classes turn out to be almost equal.

- We chose the best model for which the classification accuracy on validation data is maximum which is 57.54% when the covariance matrices are different.

- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 54.908 & 45.0920 \\ 44.9387 & 55.0613 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 58.0769 & 41.9231 \\ 51.1538 & 48.8462 \end{bmatrix}$$

2.3 Overlapping data

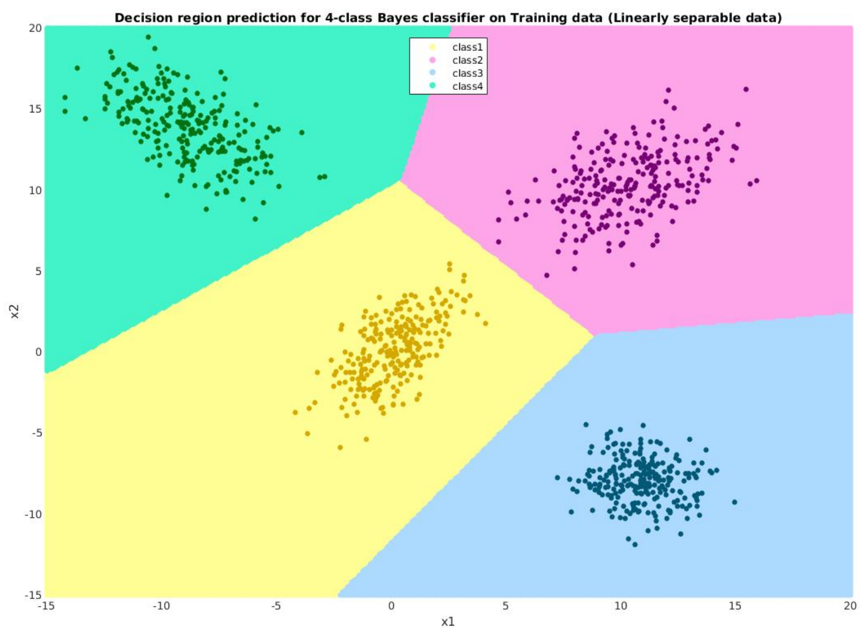

Figure 11: Best Model when the covariance matrix is same

Table 7: Classification accuracies of the model for different values of covariance matrix

Covariance matrix	**same C**	**different C**
$TrainData$	91.50	92.10
$ValidationData$	93.83	94.5

Observation:

- Classification accuracy on test data for the best model is 93%
- The decision region boundary for case I is linear whereas the decision region boundary for case II is quadratic.
- The best model is chosen for which the accuracy on validation data is maximum which is 94.50% for case II
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 92.80 & 0.40 & 3.60 & 3.20 \\ 0 & 92 & 4 & 4 \\ 2.80 & 6.40 & 90.80 & 0 \\ 2.80 & 4.40 & 0 & 92.80 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 92 & 0 & 4 & 4 \\ 0 & 92 & 5 & 3 \\ 3 & 5 & 92 & 0 \\ 3 & 1 & 0 & 96 \end{bmatrix}$$

3 Gaussian Mixture Model

3.1 GMM with Bayes

3.1.1 Linearly separable data

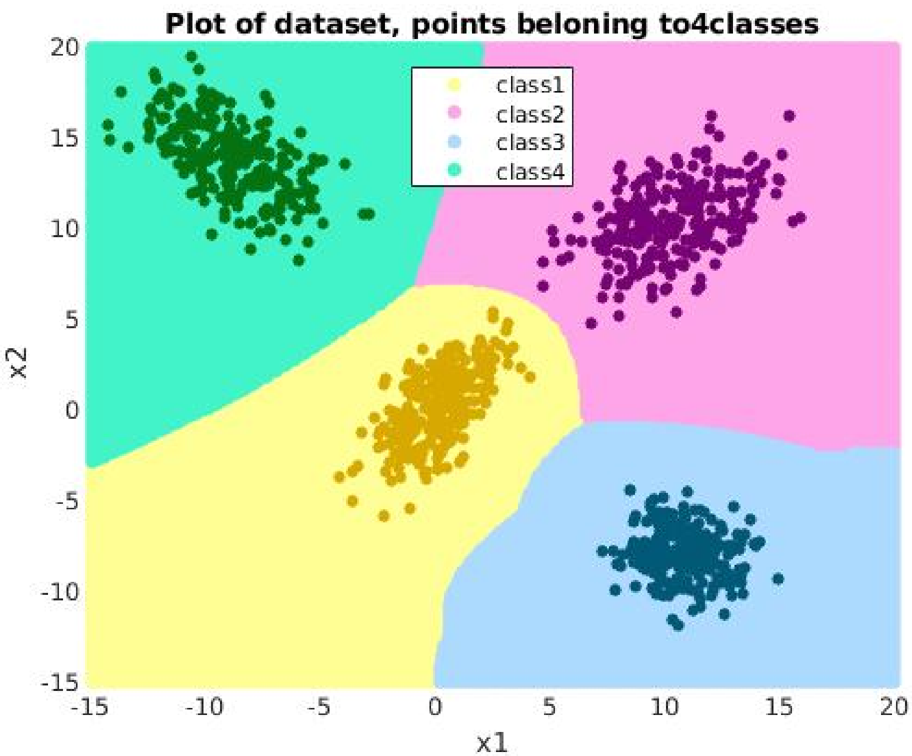

Figure 12: Decision Region plot for the Best Model with Q(No. of components)=3 for all 4 classes

Table 8: Log Likelihood of the model for different values of Hyperparameter(Q) for class 1

No of components	**Q=2**	**Q=3**	**Q=4**	**Q=5**	**Q=6**
$TrainData$	-1052	-1053	-902	-841.8	-829.8
$ValidationData$	-658.0285	-660.5841	-578.8755	-545.9248	-544.3055

Table 9: Classification accuracies of the model for different values of Hyper-parameter(Q)

No of components	**Q=2**	**Q=3**	**Q=4**
$TrainData$	100	100	100
$ValidationData$	100	100	100

Observation:

- Classification accuracy on test, train and val data for the best model is 100 for Q=2,3 and 4 (for all classes).
- The best model is chosen as Q=2 for Class1, and Q=3 for Class 2,3,4. Since Val accuracy was already 100% with Q=2(or 3), there is no need to choose higher Q for the best model.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

3.1.2 Non-Linearly separable data

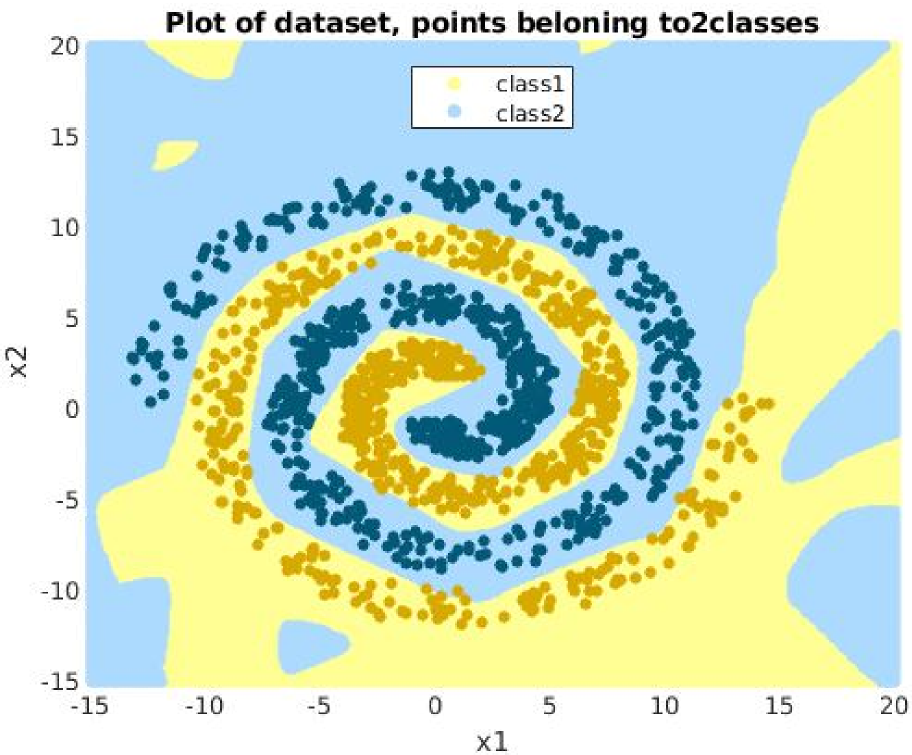

Figure 13: Decision Region plot for the Best Model with Q=10 for Class 1 and Q=9 for Class 2 for the 2 classes

Table 10: Log Likelihood of the model for different values of Hyperparameter(Q)

No of components	**Q=2**	**Q=3**	**Q=5**	**Q=6**	**Q=8**	**Q=9**	**Q=10**
$TrainData$	-6012.3	-5845.5	-4963.6	-4744.8	-4118.1	-3818.9	-3819.8
$ValidationData$	-3625.6	-3523.5	-3014.9	-2862.1	-2495.1	-2320.3	-2355.0

Table 11: Classification accuracies of the model for different values of Hyperparameter(Q)

No of components	**Q=6**	**Q=7**	**Q=10, Q=9**
$TrainData$	60.58%	69.2%	99.84%
$ValidationData$	59.23%	68.8%	99.97%

Observation:

- Classification accuracy on test data for the best model is 99.84%
- Increasing number of components increased the log likelihood, and also the accuracy on the validation set. Q=10 for Class1 and Q=9 for Class2 gave the best accuracy for the given data.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 99.6933 & 0.3067 \\ 0 & 100.00 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 98.2097 & 1.7903 \\ 0.2558 & 99.7442 \end{bmatrix}$$

3.1.3 Overlapping data

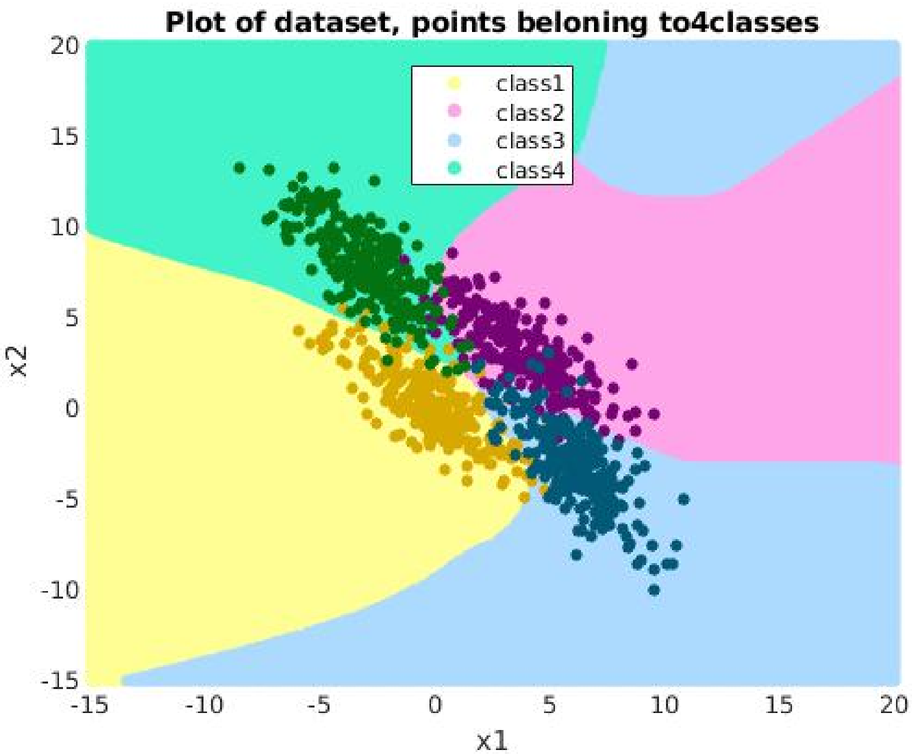

Figure 14: Decision Region for Best Model with Q=3 for all 4 classes

Table 12: Classification accuracies of the model for different values of Hyperparameter(Q)

No of components	**Q=2**	**Q=3**	**Q=4**
$TrainData$	93.2	92.10	93.2
$ValidationData$	92	93	93

$Observation$:

- Classification accuracy on test data for the best model is 92%

- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 94.0000 & 0.4000 & 2.4000 & 3.2000 \\ 0 & 92.4000 & 4.0000 & 3.6000 \\ 2.8000 & 4.4000 & 92.8000 & 0 \\ 2.8000 & 4.4000 & 0 & 92.8000 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 93 & 0 & 3 & 4 \\ 0 & 92 & 5 & 3 \\ 4 & 6 & 90 & 0 \\ 5 & 2 & 0 & 93 \end{bmatrix}$$

3.1.4 GMM with Naive Bayes

4 Non Parametric Method : K - Nearest Neighbours for PDF Estimation

In this approach, given a new sample x, we consider a small sphere centered on x at which we will be estimating the density *p(x)* and we allow the radius of the sphere to grow untill it contains exactly K data points. The estimate of *p(x)* is given by:

$$p(\mathbf{x}/y_i) = K/N_iV_i \tag{5}$$

$$i^* = argmax_i \; p(\mathbf{x}/y_i)P(y_i) \tag{6}$$

$$i^* = argmax_i \; 1/V_i \tag{7}$$

4.1 Two-Dimensional Artificial Dataset

4.1.1 Linearly Separable Data

Figure 15: Best model : 1-NN

Observations:

- For k = 1, accuracy of 100% is obtained on train, test and validation datasets.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$$

4.1.2 Non-linearly Separable Data

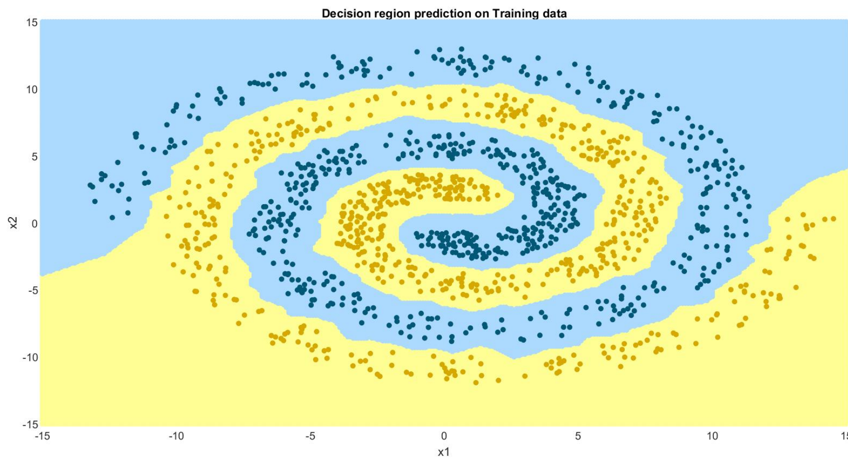

Figure 16: Best model : 1-NN

Observations:

- For k = 1, accuracy of 100% is obtained on train, test and validation datasets.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

4.1.3 Overlapping Data

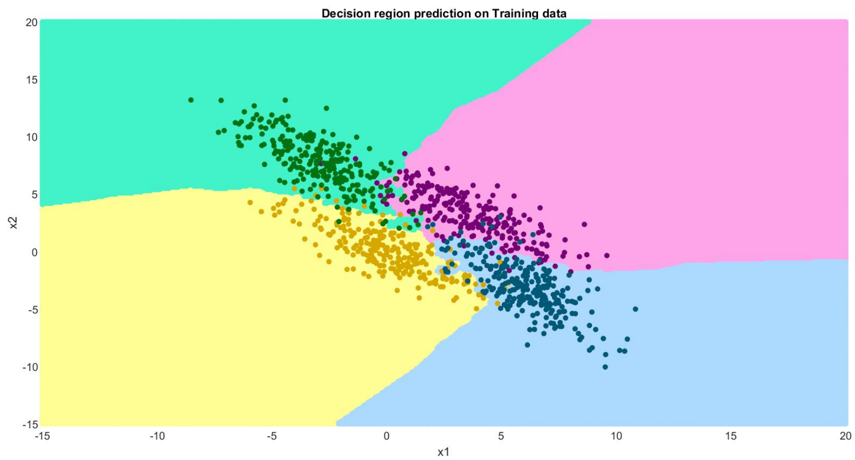

Figure 17: Best model : 3-NN

Table 13: Classification accuracies of model

k	**1**	**2**	**3**	**4**	**5**	**6**
$TrainData$	100	95	93.70	93.30	92.60	92.60
$ValidationData$	90.67	92.16	93.33	92.50	93.10	93.16

Observations:

- The best model is 3-NN model with an accuracy of 93.33% on Validation Data.
- Accuracy of best model on Test Data is 90.75%.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 93.2000 & 0.8000 & 3.2000 & 2.8000 \\ 0 & 91.6000 & 4.4000 & 4.0000 \\ 1.2000 & 4.0000 & 94.8000 & 0 \\ 2.4000 & 2.4000 & 0 & 95.2000 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 92 & 0 & 4 & 4 \\ 0 & 93 & 4 & 3 \\ 5 & 7 & 88 & 0 \\ 5 & 2 & 0 & 93 \end{bmatrix}$$

5 Real world Data : Static pattern classification

5.1 Non parametric : K Nearest Neighbour

Table 14: Classification accuracies of model

k	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
$TrainData$	100	75.59	71.82	67.79	66.79	67.79	65.03	65.03	67.16	66.41
$ValidationData$	53.98	59.29	59.28	54.86	60.17	53.98	62.83	65.48	59.29	60.17

Observations:

- The best model is 8-NN model with an accuracy of 65.48% on Validation Data.
- Accuracy of best model on Test Data is 62.71%.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 74.0741 & 20.9877 & 4.9383 \\ 35.6667 & 57.6667 & 6.6667 \\ 23.3918 & 11.6959 & 64.9123 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 70.9677 & 22.5806 & 6.4516 \\ 40.6977 & 56.9767 & 2.3256 \\ 32.6531 & 10.2041 & 57.1429 \end{bmatrix}$$

5.2 GMM : Bayes Classifier

Observations:

- GMM was run on this data set, best model to fit in the data gave Accuracy of 54.86.% on Validation set.
- Accuracy on training set : 61.76%
- Accuracy on Test set : 58.77.%
- Best Model for this data is with hyperparameter Q=2, as the data is clustered in around roughly two groups. This was verified by running serveral iterations of K-Means, very few points could be assigned to more clusters if more than 2 were tried.
- Confusion matrix(%) for best model on train data:

$$\begin{bmatrix} 82.09 & 7.71 & 10.18 \\ 57.33 & 33.33 & 9.33 \\ 26.90 & 0 & 73.09 \end{bmatrix}$$

- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 78.4946 & 13.9785 & 7.5269 \\ 63.9535 & 27.9070 & 8.1395 \\ 20.4082 & 4.0816 & 75.5102 \end{bmatrix}$$

5.3 Real world Data : Varying Length Pattern Classification

Table 15: Classification accuracies of model

k	1	2	3	4
$TrainData$	53.02	52.07	49.66	45.24
$ValidationData$	43.86	45.08	42.36	41.33

Observations:

- Out of all the models tried, the best model is 2-NN model with an accuracy of 45.08% on the Validation Dataset.
- Accuracy of best model on Test Data is 43.23%
- Confusion matrix(%) for best model on test data:

$$\begin{bmatrix} 24.2048 & 27.4787 & 11.7921 & 32.9403 & 3.5842 \\ 8.1932 & 49.3210 & 11.9975 & 25.1372 & 5.3512 \\ 8.9636 & 25.8560 & 31.0607 & 30.8477 & 3.2720 \\ 7.7884 & 22.2544 & 10.3582 & 56.6415 & 2.9576 \\ 5.4017 & 42.4284 & 8.0332 & 20.7295 & 23.4072 \end{bmatrix}$$